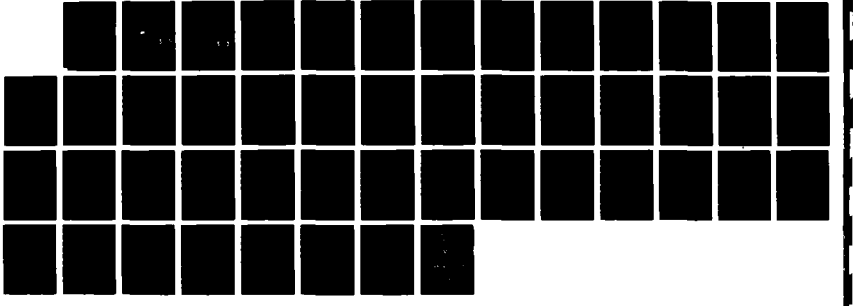
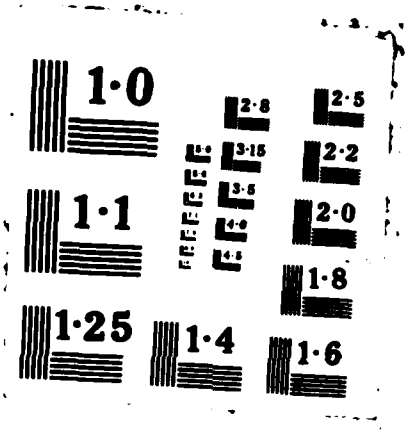
AD-A188 921 STATIC AND DYNAMIC JAMMING OF NETWORKS(U) AXIOMATIX LOS 1/1
ANGELES CA U CHENG 23 DEC 87 R8712-6 ARO-24649.2-EL-S
DAAL03-87-C-0007

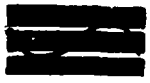
UNCLASSIFIED F/G 17/4.1 NL




1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0

STATIC AND DYNAMIC JAMMING OF NETWORKS

Interim Technical Report



Axiomatix

(2)

STATIC AND DYNAMIC JAMMING OF NETWORKS

Interim Technical Report

Unjeng Cheng

December 23, 1987

U. S. Army Research Office
P. O. Box 12211
Research Triangle Park, NC 27709-2211

Contract No. DAAL 03-87-C-0007

Axiomatix
9841 Airport Boulevard
Suite 912
Los Angeles, CA 90045

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION LIMITED

Axiomatix Report No. R8712-6

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD-A188 921

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: ARO 24649.2-EL-S	2. GOVT ACCESSION NO.: NA
3. RECIPIENT'S CATALOG NUMBER: NA	
4. TITLE (and Subtitle): STATIC AND DYNAMIC JAMMING OF NETWORKS	5. TYPE OF REPORT & PERIOD COVERED: Interim
6. PERFORMING ORG. REPORT NUMBER:	
7. AUTHOR(s): Unjeng Cheng	8. CONTRACT OR GRANT NUMBER(s): DAAL03-87-C-0007
9. PERFORMING ORGANIZATION NAME AND ADDRESS: AXIOMATIX, 9841 Airport Blvd., Suite 912, Los Angeles, CA 90045	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: NA
11. CONTROLLING OFFICE NAME AND ADDRESS: U. S. Army Research Office, P. O. Box 12211, Research Triangle Park, NC 27709-2211	12. REPORT DATE: December 23, 1987
	13. NUMBER OF PAGES: 41
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

NA

18. SUPPLEMENTARY NOTES

The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Code-Division-Multiple-Access(CDMA), Communications, Dynamic Jamming, Networks, Packet Radio, Spread Spectrum, Static Jamming,Queueing Network Analyzer (QNA)

20. ABSTRACT

The performance of a packet radio network (PRnet) under various intelligentjamming strategies is examined. Probability destiny function of various jamming strategies are derived. These are classified as dynamic or static, depending on whether or not the density function is time varying.

The investigation is carried out in two stages. The static jamming attack is considered first. The mathematical formulas are introduced and the results are compared to the simulation results. The dynamic jamming attack is addressed in the second stage. The possible observables are discussed in this report.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE 10 UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

The survivability of a packet radio network (PRnet) under jamming attack is an important issue. Conceptually, a network could be attacked on three layers of importance, namely, the network, link, and physical layers. In the game played between communicators and jammers, many factors could affect the results. The communicators have the choices of the routing algorithms, the channel quality monitoring schemes, and the network information exchange schemes. The jammers have the choices of static, dynamic, and follower jamming, as well as certain monitoring capabilities. The main thrust of this investigation is to understand the behavior of the existing PRnet under the intelligent jamming attack. Hopefully, the results of this research can lead to the design of the future generation of the PRnet.

A jammed network could be formulated as an open queueing network, and the performance measures would be the traffic intensity on each link, packet delay and queue length at each node, and the end-to-end packet delay. Many mathematical formulations and simulation models exist based on the equilibrium network analysis. With certain modifications, these formulations and models can be adapted to analyze the PRnet behavior under the static jamming attack. In the dynamic jamming environment, however, the problem becomes more complex because of the non-existence of the network equilibrium state. The ultimate goal of this study is to identify the appropriate network observables and to extend the existing theory and simulation models to the dynamic jamming environment.

The jamming attack can be described in the probabilistic manner. The event space of the jamming strategies contains the jamming topology, the jammer power level, and the jammer duty factor. Each jamming strategy is described by a probability density function over the event space. Since the jamming strategies can change from time to time, the jamming attack is a random process and the jamming probability density function (JPD) can be time-varying. The jamming attack is said to be static if the JPD does not change with time. In this case, the performance measures can be observed in the long-term average

sense. On the other hand, the dynamic jamming attack has the time-varying probabilistic descriptions. The complexity of modelling the dynamic jamming resides in the relative length of the jammer-time-constant (JTC) (the time interval between two successive jammer actions) and the communicator-time-constant (CTC) (the time interval between two successive network actions). If the JTC is very short compared to the CTC, the communicators see only the average jamming effect; thus the problem is simplified to the static jamming case. If the JTC is very long compared to the CTC, the network can reach its equilibrium state before the jammer changes its status; thus the static jamming model can be valid for each jammer action. The most complex case is when the JTC and CTC are of comparable magnitude. In this case, the long-term average may not make sense because the network may not be able to reach its equilibrium state at all. Thus, certain forms of short-term average of the performance measures must be derived.

The investigation is carried out in two stages. The static jamming attack is considered first. The mathematical formulations are introduced and the results are compared to the simulation results. The dynamic jamming attack is addressed in the second stage. The possible observables are discussed in this report and the simulation results will be presented in the next report. The mathematical formulations for the dynamic jamming threat will be addressed in the future study.

2. Network Models

The networks considered in this paper consist of geographically dispersed radio units (receivers/transmitters), where each unit can directly hear only some of the others. Therefore, multiple hops may be required for the packets to reach their destinations. All units in the network share the same radio channel by using the spread-spectrum slotted-ALOHA access protocol. Every node is equipped with a transmission buffer. The arriving packets to each node, including the external and transit packets, are stacked into the buffer and are served on the first-come-first-serve basis. A node with non-empty buffer (i.e., node is busy) at the beginning of a time slot is either transmitting or receiving in this slot; the probability of transmission is determined by the protocol. If a packet is not received successfully by the target neighbor, a retransmission is scheduled by the source node at a later time. We assume that the transmission probability at each node is independent of the number of packets in its non-empty buffer. Under this assumption, the interaction between nodes is through the busy probabilities of the nodes. Note that the busy status of the nodes forms complex random processes, which are very difficult to derive analytically. Two facts regarding the node-busy processes must be kept in mind, namely, (1) the busy processes of different nodes may be correlated, and (2) the busy status of each node may be correlated from slot to slot. Certain assumptions about these node-busy processes are usually needed to simplify the problems. For instance, Silvester and Lee assumed these processes being the node-independent Bernoulli processes, namely, the busy status of the nodes is slot-by-slot and node-by-node independent [1]. We also see that the always-busy assumption [2-3] gives an upper bound on the interaction between nodes. Certainly, more sophisticated models and bounds for the busy processes do exist. However, we do not address them in this report.

The spread-spectrum multiple-access can be accomplished by the common-code, the transmitter-based code, or the receiver-based code technique. For the common-code case, all radio units use the same spread-spectrum code. For the transmitter-code case,

every radio unit has its distinct spread-spectrum transmission code. At the beginning of each transmission, the radio unit must broadcast its code identification to all its neighbors through a common code channel. On the other hand, each node can listen to one and only one code among all the codes identified in the common-code channel. For the receiver-based code case, every radio unit listens to its unique spread-spectrum receiving code. Thus a transmitting radio must transmit with the receiving code of the target neighbor. The common code and the transmitter-based code multiple-access have a common property, namely, a node may listen to a packet which is not addressed to it. On the other hand, this situation does not happen in the receiver-based code multiple-access. It is seen later that the computation of the successful packet reception probability in the receiver-based code case is more complex than that in the common code and the transmitter-based code cases.

Let the nodes of the network be represented by the integers 1, 2, . . . , N. The neighbors of node i are the nodes which can hear node i. The set of all neighbors of node i is denoted by X_i. The number of nodes in X_i is denoted by N_i. The external traffic arriving to node i is assumed to be the Bernoulli process. The mean arrival rate of the external traffic from the source node s to the destination node d is denoted by λ_{sd}. The mean external arrival rate to node i, denoted by λ_i, is given by

$$\lambda_i = \sum_{d=1}^{N} \lambda_{id} \quad . \tag{2.1}$$

In the above equation, $\lambda_{ii} = 0$ for $1 \leqq i \leqq N$ are assumed. The departure (packets leaving the network) rate from node i, denoted by D_i, is given by

$$D_i = \sum_{s=1}^{N} \lambda_{si} \quad . \tag{2.2}$$

The successful transit (packets to be forwarded to the next nodes) arrival rate to node i is denoted by s_i. The composite arrival rate to node i is denoted by Γ_i. We have $\Gamma_i = \lambda_i + s_i$. The performance measures are the traffic intensity, the mean packet delay, and the mean

queue size at each node, as well as the network queue size and the network packet delay. Let D_i and Q_i denote the mean packet delay and the mean queue size at node i, respectively. The network mean queue size, denoted by Q, is given by

$$Q = \sum_{i=1}^{N} Q_i \quad . \tag{2.3}$$

The total external packet arrival rate to the network, denoted by λ, is given by

$$\lambda = \sum_{i=1}^{N} \lambda_i \quad . \tag{2.4}$$

The average network packet delay, denoted by D, is computed from Q and λ using the Little's result, namely, $D = Q/\lambda$.

3. JAMMING MODELS

The dynamic jamming strategies can be described by the random vector (JSR_1, $JSR_2, \ldots, JSR_N$) and the associated probability density function f (JSR_1, $JSR_2, \ldots,$ JSR_N, t), where JSR_i is the jammer-to-signal power ratio at node i and t is the time measured in slot. We see that the jamming status at different nodes can be correlated in the general scenario, and the jamming-probability-density (JPD) function f(JSR_1, $JSR_2, \ldots,$ JSR_N, t) can also be correlated from slot-to-slot. The jamming attack is said static if the JPD does not vary with time.

3.1 Static Jamming Models

The static jamming strategies are defined by their JPD f(JSR_1, $JSR_2, \ldots, JSR_N$). A special case is the node-independent (NI) static jamming, whose JPD can be written as

$$f(JSR_1, JSR_2, \ldots, JSR_N) = \prod_{i=1}^{N} f_i(JSR_i) \quad . \tag{3.1}$$

The NI static jamming can be further simplified by assuming that JSR_i can take only two values, namely, zero if the jammer is off and a fixed non-zero value if the jammer is on. This jamming strategy is referred to as the on-off NI static jamming. The theory presented in Section 4 is for the on-off NI static jamming. It can be extended easily to the NI static jamming case.

3.2 Dynamic Jamming Models

The dynamic jamming attack can be a complex random process. In this report, we consider only the periodic fixed block jammer (PFBJ). In this case, the jammer follows a fixed time-domain pattern in each period and the jamming strategy is fixed in each block of consecutive time slots. This jamming attack is delineated in Figure 1. A special case of the PFBJ is to alternate between two jamming strategies. One strategy is on for T_1 slots of

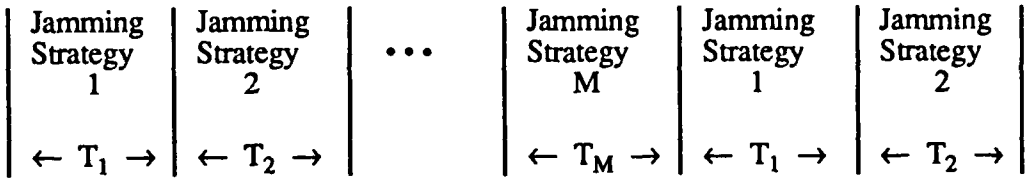


Figure 1. Periodic Fixed Block Jammer

interval and the other is on for T_2 slots of interval. This form of dynamic jamming is referred to as the two-stage periodic dynamic jamming. If T_1 and T_2 are small, the network sees the average jamming effect of these two jamming strategies. If T_1 and T_2 are so large that the network can reach its equilibrium state, then the static jamming model can be applied to analyze the network under each jamming strategy. Conceptually, the network shows different behavior in four time intervals, which are shown in Figure 2.

	Jamming Strategy 1	Jamming Strategy 1	Jamming Strategy 2	Jamming Strategy 2	
...	Routing Table best for Jamming Strategy 2	Routing Table best for Jamming Strategy 1	Routing Table best for Jamming Strategy 1	Routing Table best for Jamming Strategy 2	...

Figure 2. Network Behavior under the Two-Stage Periodic Dynamic Jamming

4. STATIC JAMMING ATTACK

For the static jamming attack, no network adaptability is required. Here we are interested in the performance of a fixed routing table under the jamming. Several problems can be addressed. The first problem is to develop the accurate mathematical methods for analysis of the static-jammed PRnets. The second problem is to find the best routing table under the worst-case jamming. Thirdly, since the optimum routing tables in the jammed and non-jammed situations can be different, it is interesting to see how much difference between the traffic distributions under the jammed and non-jammed environments. Note that the traffic distribution can be a useful observable for the follower jammers. The results of this study can be useful for the understanding of the detectability of the adaptive routing algorithms, which attempt to build the best routing table for the prevailing environment. The possibility of extending the theory for the static jamming attack case to the periodic fixed block jamming case will be examined in the future report.

The static jamming models were described in Section 3.1. Several important concepts regarding the networks with the fixed routing table are introduced in this section. Two analytical methods are discussed in Section 4.1. The simulation model is delineated in Section 4.2.

The routing table can be expressed in terms of a set of parameters δ_{sdij} that specify what fraction of the (s,d) traffic uses the link (i,j). The traffic flow through the link (i,j) can be calculated by

$$f_{ij} = \sum_{s,d} \lambda_{sd} \delta_{sdij} \quad . \tag{4.1}$$

The probability that a packet will take the $i \rightarrow j$ path upon successfully leaving node i is denoted by θ_{ij} and it can be computed by

$$\theta_{ij} = \frac{f_{ij}}{\sum_{\ell \varepsilon x_i} f_{i\ell}} \quad . \tag{4.2}$$

The composite arrival rate, the departure rate, and the link flow at node i are related by

$$\Gamma_i + D_i = \sum_{j \in x_i} f_{ij} \quad . \quad . \quad . \quad (4.3)$$

Equation (4.3) is used to compute the composite arrival rate. It is seen later that Γ_i plays an important role in the network equilibrium analysis.

Another important parameter is the probability of successful reception. A packet arriving at node j may not be received correctly because of (1) the transmission of the target node, (2) the multiple-access interference, (3) the jamming interference, and (4) the thermal noise. The probability of successful reception given an $i \rightarrow j$ transmission is denoted by $P\{s|i \rightarrow j\}$. The explicit expression for $P\{s|i \rightarrow j\}$ depends on the multiple-access schemes, the jamming formats, and the modulation and coding formats. For the transmitter-based code multiple access and on-off jamming, we have

$$P\{s|i \rightarrow j\} = (1 - p_j b_j) \sum_{k=1}^{N_j} \frac{1}{k} \text{Prob}\{k|i \rightarrow j\} \left[(\gamma_j P_{Aj}(k,J) + (1 - \gamma_j) P_{Aj}(k,NJ) \right] \quad (4.4)$$

where

$$b_j = \text{the busy probability at node j,}$$

$$p_j = \text{the transmission probability at node j,}$$

$$\text{Prob}\{k|i \rightarrow j\} = \text{Prob}\{\text{k packets are heard at node j}|i \rightarrow j\}\ , \quad (4.5)$$

$$P_{Aj}(k,J) = \text{Prob}\{\text{successful reception}|\text{k packets are heard at node j and the jammer is on}\}\ , \quad (4.6)$$

$$P_{Aj}(k,NJ) = \text{Prob}\{\text{successful reception}|\text{k packets are heard at node j and the jammer is off}\}\ , \quad (4.7)$$

$$\gamma_j = \text{Prob}\{\text{the jammer is on at node j}\}. \quad (4.8)$$

The factor (1/k) is present because that node j has the probability (1/k) to pick up the packet from node i among the k packets heard by it. Assuming that the node-busy processes are

independent from node to node, Prob{k|i→j} can be computed by expanding the following polynomial product:

$$\prod_{\substack{\ell \in X_j \\ \ell \neq i}} (p_\ell \; b_\ell \; x + (1 - p_\ell \; b_\ell)) \; . \tag{4.9}$$

Explicitly, Prob{k|i→j} is the coefficient of the x^{k-1} term in the above expansion.

For the receiver-based code multiple-access and on-off jamming, we have

$$P\{s|i\rightarrow j\} = (1 - p_j b_j) \sum_{k=1}^{N_j} (\gamma_j P_{A_j}(k,J) + (1 - \gamma_j) P_{A_j}(k,NJ)) \; x \left[\sum_{n=1}^{k} \frac{1}{n} \text{Prob}\{n,k|i\rightarrow j\} \right] \tag{4.10}$$

where $P_{A_j}(k,J)$ and $P_{A_j}(k,NJ)$ are defined as before, and

$$\text{Prob}\{n,k|i\rightarrow j\} = \text{Prob}\{\text{k packets are heard at node j and n packets are targeted at it}|i\rightarrow j) \; . \tag{4.11}$$

The computation of Prob{n,k|i→j} needs the information about the link flow distribution. We delay its derivation to Section 4.1. Comparing equations (4.4) and (4.10), we see that the computation of the successful reception probability in the receiver-based code case is more complex than the transmitter-based code case. In Section 4.1, we show that finding the busy probabilities through equation (4.4) needs only one-dimensional iterations; whereas, finding the busy probabilities through equation (4.10) needs two-dimensional iterations.

Since the main interest of this investigation is the network aspect of the jamming threat, we assume the simplest signal format and jamming format, namely, the continuous tone jamming and the coded direct-sequence binary-phase-shift-key modulation. We have

$$P_{A_j}(k,\Lambda) = \sum_{\ell=0}^{e} \binom{L}{\ell} P_{CE_j}^{\ell}(k,\Lambda) \, [1 - P_{CE_j}(k,\Lambda)]^{L-\ell} \tag{4.12}$$

for the e-error correction code of block length L (it is also the packet length here). We also let $\Lambda = J$ if node j is jammed, and let $\Lambda = NJ$ if node j is not jammed. $P_{CEj}(k,\Lambda)$ is the channel symbol error probability given by:

$$P_{CEj}(k,\Lambda) = Q\left[(2E_{cs}/N_{eq}(k,j,\Lambda))^{1/2}\right] \tag{4.13}$$

where

$$\frac{E_{cs}}{N_{eq}(k,j,J)} = \left[\left(\frac{E_{cs}}{N_0}\right)^{-1} + \frac{(k-1)\alpha}{\eta} + \frac{JSR_j}{\eta}\right]^{-1} \tag{4.14}$$

and

$$\frac{E_{cs}}{N_{eq}(k,j,NJ)} = \left[\left(\frac{E_{cs}}{N_0}\right)^{-1} + \frac{(k+1)\alpha}{\eta}\right]^{-1} \tag{4.15}$$

and

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-x^2/2}\,dx \ .$$

In the above equations $\eta = T_{cs}/T_c$ is the spreading ratio of the channel symbols, E_{cs}/N_0 is the signal-energy-to-thermal-noise ratio per channel symbol, and α is the multiple-access coefficient depending on the cross correlation between the particular multiple-access codes in use. In this report, the numerical results are computed for $\alpha = 1$. For the uncoded case, equation (4.12) can be simplified by letting $e = 0$. We have

$$P_{Aj}(k,\Lambda) = \left[1 - P_{CEj}(k,\Lambda)\right]^L \ . \tag{4.16}$$

The numerical results presented in this report (Section 4.1.2) are for the uncoded case.

Assuming the Bernoulli-node-busy processes, the mean service time of a packet from node i to node j, denoted by τ_{ij}, can be computed from the probability of successful reception, namely:

$$\tau_{sij} = 1/\left(p_i P\{s|i\rightarrow j\}\right) \ . \tag{4.17}$$

The mean service time of a packet transmitted by node i, denoted by τ_{si}, can be expressed as

$$\tau_{si} = \sum_{j \epsilon X_i} \theta_{ij} \tau_{sij} \quad . \tag{4.18}$$

The probability of successful reception of a packet transmitted by node i, denoted by $P\{s|i\}$, can be computed from $P\{s|i \rightarrow j\}$ by

$$P\{s|i\} = \sum_{j \epsilon X_i} \phi_{ij} P\{s|i \rightarrow j\} \tag{4.19}$$

where

$$\phi_{ij} = \frac{f_{ij} \tau_{sij}}{\sum_{\ell \epsilon X_i} f_{i\ell} \tau_{si\ell}} \quad . \tag{4.20}$$

It is interesting to verify that

$$\tau_{si} = 1/\left(p_i P\{s|i\}\right) \quad . \tag{4.21}$$

4.1 Analytical Methods

The analytical methods are derived based on the algorithm by Silvester and Lee [1], the algorithm by Su [4], and the algorithms in the Queueing Network Analyzer (QNA) by Whitt [5]. Briefly speaking, the algorithms in [1] and [4] are generalized and are used to compute the busy probability at each node, and the QNA is used to compute the mean packet delay and the mean queue size at each node.

For the multiple-access protocol described in Section 2, the interaction between nodes is through the busy probabilities of the nodes. As mentioned before, the node-busy processes are usually too complex to be derived analytically. In the subsequent derivation, we assume the node-independent Bernoulli busy processes. This assumption is needed in order to use equations (4.9) and (4.17). The busy probabilities of the nodes can be determined by solving the associated queueing model and the behavior of the node-buffers can then be analyzed using the QNA.

A general result in the queueing theory is that the busy-probability is equal to the product of the mean service time and the composite arrival rate [5], namely,

$$b_i = \Gamma_i \tau_{si} = \Gamma_i / (p_i P\{s|i\}) \quad . \tag{4.22}$$

Note that $P\{s|i\}$ is a function of the busy probabilities. Equations (4.4) through (4.22), for $1 \leqq i \leqq N$, form a set of non-linear equations. Its solution can be found using the iteration methods, which are different for the transmitter-based code and the receiver-based code cases.

Transmitter-Based Code Case

For the transmitter-based code case, equations (4.4), (4.9), (4.17), (4.18), and (4.22) can be solved by the algorithm suggested by Silvester and Lee [3]; thus the busy probabilities and the mean service time can be determined. The algorithm in [1] is modified as follows:

Algorithm 1:

Step 1: Let $b_i = 0$ for $1 \leqq i \leqq N$.

Step 2: Compute $P\{s|i \rightarrow j\}$ for $1 \leqq i \neq j \leqq N$ using equation (4.4).

Step 3: Compute τ_{si} for $1 \leqq i \leqq N$ using equation (4.18).

Step 4: Let $a_i = \Gamma_i \tau_{si}$.

Step 5: If $|(a_i - b_i)/a_i| < \delta$ (δ is the tolerance factor), then stop; otherwise,

let $b_i \leftarrow a_i$ and go to step 2.

The only assumption made in the derivation of the above algorithm is the Bernoulli node-independent busy processes. No assumption is made regarding the arrival and the service processes at each node. Therefore, the result presented here is more general than that in [1]. Combining Algorithm 1 with the QNA, we get a static-jammed network analyzer for the transmitter-based code case.

Receiver-Based Code Case:

For the receiver-based code case, equations (4.10), (4.17), (4.19), (4.20), (4.21), and (4.22) are solved by the algorithm suggested by Su [4]. Note that Prob{n,k|i→j} in equation (4.10) can be computed by expanding the following polynomial product:

$$\prod_{\substack{\ell \in X_j \\ \ell \neq i}} [p_\ell \, b_\ell \, \phi_{\ell j} \, xy + p_\ell \, b_\ell \, (1 - \phi_{\ell j})x + (1 - p_\ell \, b_\ell \,)] \quad . \tag{4.23}$$

Explicitly, $P\{n,k|i\rightarrow j\}$ is the coefficient of the $x^{k-1}y^{n-1}$ term in the above expansion. Given the busy probabilities, the computation of $P\{s|i\rightarrow j\}$ needs $\phi_{\ell j}$, $\ell \in X_j$ (equations (4.10) and 4.23)), whose computation, in turn, needs $P\{s|i\rightarrow j\}$ (equation (4.20)). Thus, equations (4.10), (4.20), and (4.23) form a set of non-linear equations, which can be solved by the iteration method. The solution $P\{s|i\rightarrow j\}$ is a function of the busy probabilities. Together with equation (4.22), they form another set of non-linear equations and can be solved by the second iteration process. The algorithm in [4] is modified as follows:

Algorithm 2:

Step 1: Let $b_i = 0$ for $1 \leqq i \leqq N$ and let $\phi_{ij} = f_{ij}$ for $1 \leqq i, j \leqq N$.

Step 2: Compute $P\{s|i\rightarrow j\}$ for $1 \leqq i \neq j \leqq N$ using equation (4.10).

Step 3: Evaluate the right hand side of equation (4.20) and let the result be denoted as Φ_{ij} for $1 \leqq i, j \leqq N$.

Step 4: If $|(\Phi_{ij} - \phi_{ij})/\Phi_{ij}| < \delta'$ (δ' is the tolerance factor) for $1 \leqq i, j \leqq N$, then go to step 5; otherwise, let $\phi_{ij} \leftarrow \Phi_{ij}$ and go to step 2.

Step 5: Compute τ_{si} for $1 \leqq i \leqq N$ using equation (4.18).

Step 6: Let $a_i = \Gamma_i \, \tau_{si}$.

Step 7: If $|(a_i - b_i)/a_i| < \delta$ (δ is the tolerance factor), then stop; otherwise, let $b_i \leftarrow a_i$ and go to step 2.

The only assumption made in the derivation of the above algorithm is the Bernoulli node-independent busy processes. No assumption is made regarding the arrival and the service processes at each node. Therefore, the result presented here is more general than that in [4]. Combining Algorithm 2 with the QNA, we get a static-jammed network analyzer for the receiver-code case.

4.1.1 QNA Algorithms

The basic idea behind the QNA algorithm is to consider each node buffer as a G/G/1 queue. There are approximation formulas for the mean queue size Q_i and the mean packet delay D_i for the G/G/1 queue case, which need only the first and second moments of the arrival and the service processes. In the original QNA model [5], the first and second moments of the service processes are assumed to be known. Thus only the first and second moments of the arrival processes must be derived. In our case, however, the first and second moments of the service processes are not known because of the multiple-access interaction and the half-duplex nature of the radio channel. With the assumption of the Bernoulli node-independent busy processes, the service process at each node can be described as the mixture of several Bernoulli processes, each process has a different mean service time, explicitly, the mean service time of the $i \rightarrow j$ packet is τ_{sij}. Thus the first and second moments of the service processes can be computed after the busy probabilities are determined.

Let σ_{si}^2 denote the variance of the service time at node i. Let τ_{ai} and σ_{ai}^2 denote the mean interarrival time and the variance of the interarrival time of the external arrival process at node i, respectively. The squared coefficients of variances of the arrival and the service processes are defined as $c_{si}^2 = \sigma_{si}^2/\tau_{si}^2$ and $c_{ai}^2 = \sigma_{ai}^2/\tau_{ai}^2$, respectively. The mean packet delay D_i is given by

$$D_i \approx \tau_{si}\left[1 + \frac{b_i(c_{ai}^2 + c_{si}^2)\, g}{2(1 - b_i)}\right] \tag{4.24}$$

where

$$g = \begin{cases} \exp\left[-\frac{2(1-b_i)}{3b_i}\frac{(1-c_{ai}^2)^2}{c_{ai}^2 + c_{si}^2}\right], & \text{if } c_{ai}^2 < 1 \\ 1, & \text{if } c_{ai}^2 \geqq 1 . \end{cases} \tag{4.25}$$

The mean queue size Q_i is given by

$$Q_i = \Gamma_i D_i \quad . \tag{4.26}$$

In order to use equation (4.24), we need τ_{si}, c_{ai}^2, and c_{si}^2. The mean service time τ_{si} can be computed by equation (4.18). The squared coefficients of variances for the service processes can be computed by

$$c_{si}^2 = \frac{1}{\tau_{si}^2}\sum_{j \varepsilon X_i} \theta_{ij}\frac{1 - p_i P\{s|i\rightarrow j\}}{\left(p_i P\{s|i\rightarrow j\}\right)^2} \quad . \tag{4.27}$$

The squared coefficients of variances for the arrival processes must be computed by the technique described in [5]. It is tailored for our need and is summarized as follows: the squared coefficients of variances c_{ai}^2, $1 \leqq i \leqq N$, are computed by solving the following set of linear equations:

$$c_{aj}^2 = \Theta_j + \sum_{i=1}^{N} c_{ai}^2 \beta_{ij} \quad , \quad 1 \leqq j \leqq N \tag{4.28}$$

where

$$\Theta_j = 1 + w_j\left\{(\psi_{0j}c_{0j}^2 - 1) + \sum_{i=1}^{N}\psi_{ij}\left[1 + \theta_{ij}(1 - D_{ij})(b_i^2\, x_i - 1)\right]\right\},$$

$$\beta_{ij} = w_j\, \psi_{ij}\, \theta_{ij}\,(1 - D_{ij})\,(1 - b_i^2),$$

$$x_i = \max\{c_{si}^2\,,\, 0.2\},$$

$$w_j = \{1 + 4(1 - b_j)^2 (\upsilon_j - 1)^{-1} \quad ;$$

$$\upsilon_j = \left[\sum_{i=0}^{N} \psi_{ij}^2 \right]^{-1} \quad ,$$

D_{ij} = Prob{a packet, upon leaving node i, will be absorbed by (finally destined for) node j}, (4.29)

ψ_{ij} = Prob{a packet arriving at node j that comes from node i}, (4.30)

ψ_{0j} = Prob{a packet arriving at node j that is an external packet}, (4.31)

c_{0j}^2 = the squared coefficient of variance of the external arrival process at node j. (4.32)

Since the external arrival processes are the Bernoulli processes, we have $c_{0j}^2 = 1 - \lambda_j$.

We see that the QNA is a method based on the first and second moments of the arrival and service processes. The advantage of this approach is that no assumptions are needed for the arrival and service processes. Therefore, the results are general. In the next section, we compare the QNA with the model derived by Silvester and Lee (SL model) [1].

4.1.2 Comparison of QNA and SL Models

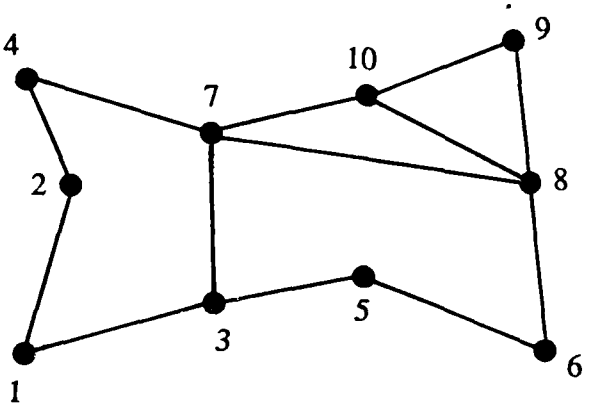
The derivation in [1] is for the unspread slotted-ALOHA multiple-access. However, it can be extended to the spread-spectrum slotted-ALOHA multiple-access. In [1], Algorithm 1 was derived by assuming both the arrival and the service processes are Bernoulli. As we mentioned before, this assumption is not always correct. Actually, the service process at each node is the mixture of several Bernoulli processes, each with different mean service time. Note that the service process is indeed Bernoulli if all its component Bernoulli processes have the same mean service time. The service process is further away from the Bernoulli process if the differences among the mean service times of its component Bernoulli processes increase. Unfortunately, this situation happens

definitely in the jamming environment, where one neighbor can be jammed worse than another. Therefore, the QNA model can be more accurate for case with the geographically dispersed static jamming attack.

Consider the 10-node network shown in Figure 3, which was used in [1]. In Figure 4, we compare the QNA and SL models for the unspread slotted-ALOHA multiple-access case using the network of Figure 3. We see the excellent agreement between two models when the network traffic is not heavy. The disagreement increases as the traffic increases. Note that both methods predict too small mean packet delay when the network is running close to the saturation point. The primary error is due to the assumption of the Bernoulli node-independent busy processes. This assumption is clearly incorrect. For instance, a queue containing ten packets in the current time slot will definitely be busy in at least the next nine time slots. Therefore, the busy processes cannot be Bernoulli. They are further away from the Bernoulli processes as the network is running closer to the saturation point. A second-order error is due to the way the node queue is modeled and solved. For the QNA, it is the approximation error in equations (4.24) and (4.28). For the SL model, it is due to the assumption of the Bernoulli arrival and service processes.

In Figure 5, we compare the QNA and SL models for the spread-spectrum slotted-ALOHA multiple-access case. As we expect, the spread-spectrum multiple-access does improve the mean network packet delay at the expense of the wider bandwidth. The observations made for Figure 4 are also valid here.

The advantage of the QNA model over the SL model is that no assumption was made for the arrival and service processes. In order to see this advantage, let us consider the 3-node network shown in Figure 6. Node 1 has traffic going to node 2 and 3, and nodes 2 and 3 have traffic going to node 1. For this simple network, there is no transit traffic at each node; thus, the arrival processes are Bernoulli. Nodes 2 and 3 have only one out-going link; thus, their service processes are also Bernoulli if the busy process at node 1 is independent of the busy processes at nodes 2 and 3. Since the assumption of the



$$[p_i] = [0.33, 0.33, 0.58, 0.33, 0.22, 0.14, 0.47, 0.38, 0.38. 0.13]$$

$$[\lambda_{ij}] = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u & 0 \\ 0 & 0 & 0 & 0 & u & 0 & 0 & u & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & u & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & u & 0 & 0 & 0 & 0 \\ 0 & u & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u \\ 0 & 0 & 0 & u & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & u & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & u & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ u & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u & 0 \end{matrix}$$

Figure 3. A 10-node network

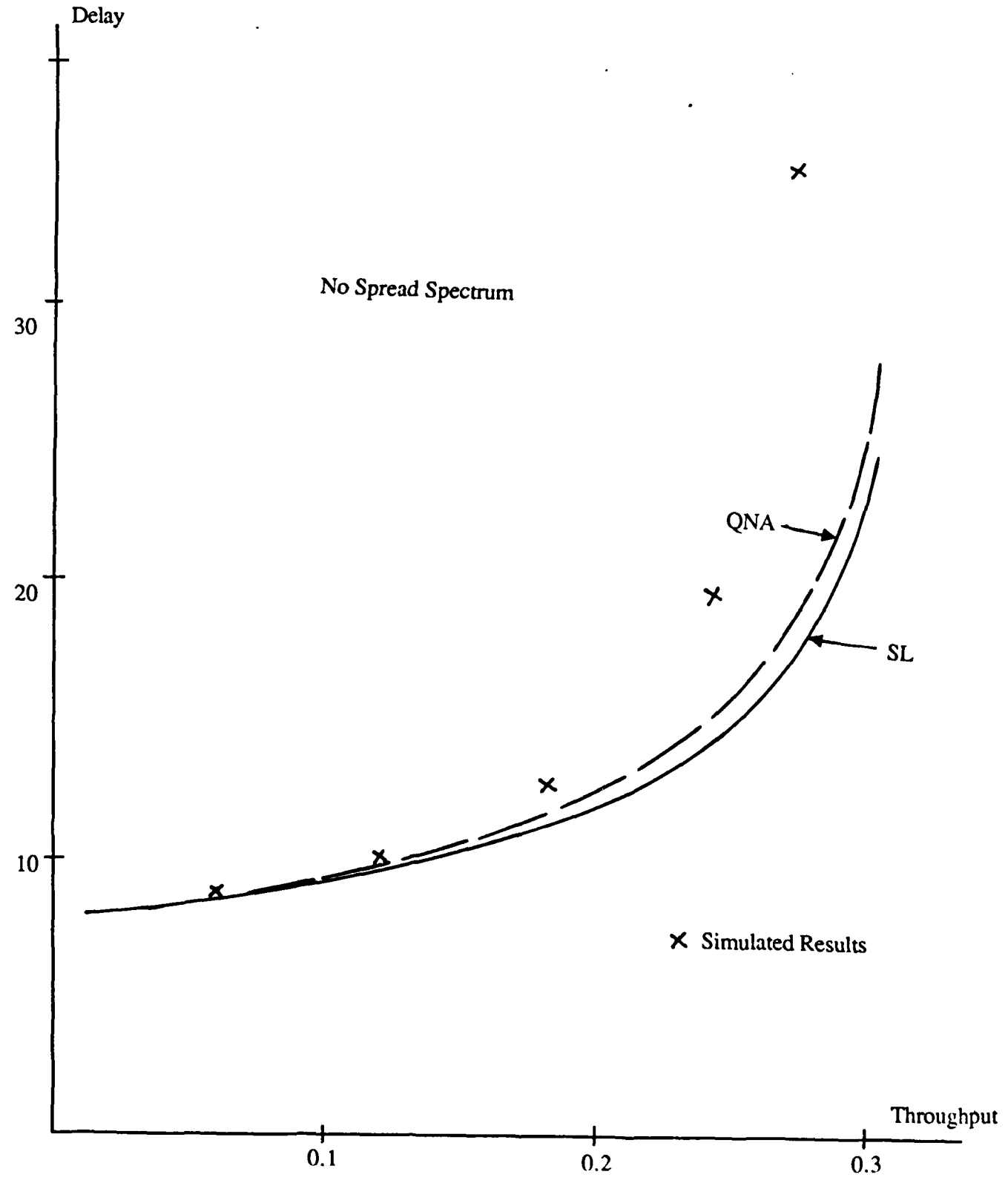


Figure 4. Comparison of the QNA and SL models for the unspread slotted-ALOHA multiple-access case. The 10-node network in Figure 3 is used.

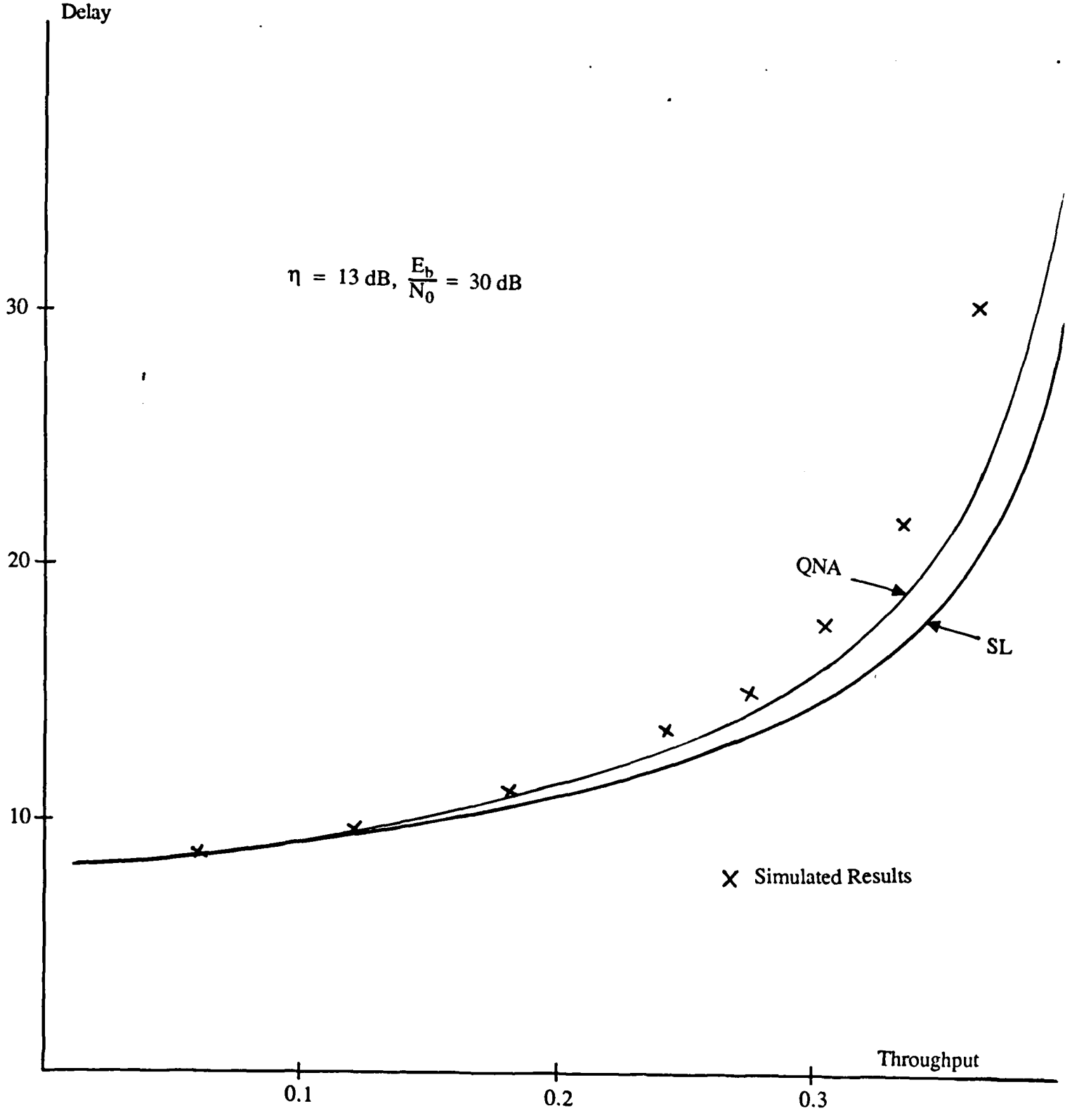
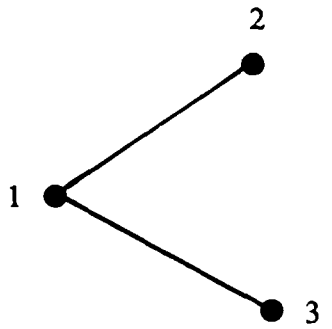


Figure 5. Comparison of the QNA and SL models for the uncoded transmitted-code spread-spectrum slotted-ALOHA multiple-access case. The 10-node network in Figure 3 is used.



$p_1 = p_2 = p_3 = 0.2$

$$[\lambda_{ij}] = \begin{matrix} 0 & u & v \\ (u+v)/2 & 0 & 0 \\ (u+v)/2 & 0 & 0 \end{matrix}$$

Figure 6. A 3-node network

Bernoulli node-independent busy processes is made for both the QNA and SL models, they should predict the same mean packet delay at nodes 2 and 3. The service process at node 1 is the mixture of two Bernoulli processes, one for the $1 \rightarrow 2$ packets and the other for the $1 \rightarrow 3$ packets. If there is no jamming and the busy probabilities at nodes 2 and 3 are roughly equal (it is probably true here since $\lambda_{31} = \lambda_{21}$ and $p_2 = p_3$), these two Bernoulli processes have roughly equal mean service time, namely, $\tau_{s12} = \tau_{s13}$, and the service process at node 1 is nearly Bernoulli. Thus the QNA and SL models should also predict the same mean packet delay at node 1 no matter how heavy the network traffic is. This observation is shown in Table 1. The same argument also holds if nodes 2 and 3 are jammed identically. This can be seen in Tables 2 and 3. We conclude that the SL model is accurate as long as the assumption of the Bernoulli arrival and service processes holds. In Tables 1 through 3, we also show that both the QNA and SL models predict the mean packet delay smaller than that obtained by simulation when the network traffic is heavy. Now let us consider the case that node 3 is jammed. The mean service time of the $1 \rightarrow 3$ packets can be significantly longer than that of the $1 \rightarrow 2$ packets. The difference between τ_{s12} and τ_{s13} increases as JSR_3 (jammer-power-to-signal-power ratio at node 3) increases. Note that the mean service time τ_{s1} is identical in the QNA and SL models; it is given by (4.18) in both cases. In the SL model, the service process at node 1 is assumed to be Bernoulli with the mean service time τ_{s1}. Note that the mean packet delay comprises two components, namely, the mean waiting time, which is the average time the packets spend in the queue, and the mean service time, which is the average time the packets take to be served. The QNA and SL models predict different mean waiting time. If the traffic is light, the mean service time is longer than the mean waiting time. On the other hand, if the traffic is heavy, the mean service time is shorter than the mean waiting time. Therefore, the difference of the mean packet delay predicted by the two models increases with the network traffic. This is shown in Tables 4 through 6. We also see that the difference between two models in the heavy traffic conditions increases as JSR_3 increases.

Throughput	Packet Delay								
	SL			QNA			Simulation		
	Node 1	Node 2	Network	Node 1	Node 2	Network	Node 1	Node 2	Network
0.04	5.51	5.35	5.43	5.56	5.38	5.47			
0.2	10.17	7.68	8.93	10.42	7.82	9.12	10.33	7.76	9.04
0.3	27.30	11.01	19.16	27.67	11.24	19.45	27.87	11.16	19.50
0.34	107.21	13.51	60.36	107.64	13.77	60.70	123.42	13.67	68.56
0.35	448.03	14.34	231.19	448.47	14.61	231.54			
0.352	1251.30	14.52	632.91	1251.74	14.80	633.27			

Table 1. Comparison of the QNA and SL models for the uncoded transmitter-based code spread-spectrum slotted-ALOHA multiple-access. The 3-node network in Figure 6 is used. η = 13 dB, E_b/N_0 = 30 dB, $JSR_1 = JSR_2 = JSR_3$ = 0 dB, u = v.

Throughput	Packet Delay								
	SL			QNA			Simulation		
	Node 1	Node 2	Network	Node 1	Node 2	Network	Node 1	Node 2	Network
0.04	10.29	5.44	7.86	10.37	5.46	7.92	10.29	5.46	7.87
0.2	98.91	8.78	53.84	99.33	8.93	54.13	100.58	8.84	54.70
0.208	189.98	9.08	99.53	190.43	9.24	99.84	193.08	9.14	101.10
0.216	2768.14	9.41	1388.78	2768.61	9.58	1389.09			

Table 2. Comparison of the QNA and SL models for the uncoded transmitter-based code spread-spectrum slotted-ALOHA multiple-access. The 3-node network in Figure 6 is used. η = 13 dB, E_b/N_0 = 30 dB, $JSR_1 = JSR_2 = JSR_3$ = –7.3 dB, u = v.

Throughput	Packet Delay								
	SL			QNA			Simulation		
	Node 1	Node 2	Network	Node 1	Node 2	Network	Node 1	Node 2	Network
0.002	338.79	5.28	172.03	338.92	5.28	172.10	342.77	5.34	175.72
0.004	512.76	5.60	259.18	513.02	5.60	259.31	509.76	5.67	256.70
0.006	1056.45	5.95	531.20	1056.83	5.95	531.39	1063.43	5.97	536.90
0.0072	2913.40	6.18	1459.79	2913.85	6.19	1460.02			
0.0076	7044.97	6.26	3525.62	7045.45	6.27	3525.86			

Table 3. Comparison of the QNA and SL models for the uncoded transmitter-based code spread-spectrum slotted-ALOHA multiple-access. The 3-node network in Figure 6 is used. η = 13 dB, E_b/N_0 = 30 dB, $JSR_1 = 0$, $JSR_2 = JSR_3$ = –5.5 dB, u = v.

Throughput	Packet Delay								
	SL			QNA			Simulation		
	Node 1	Node 2	Network	Node 1	Node 2	Network	Node 1	Node 2	Network
0.04	7.80	5.39	6.60	7.90	5.42	6.66	7.82	5.36	6.60
0.1	10.19	6.15	8.17	10.49	6.22	8.36			
0.2	23.48	8.19	15.84	24.27	8.34	16.41	24.89	8.27	16.57
0.24	54.79	9.54	32.17	57.11	9.73	33.42	59.59	9.58	34.59
0.26	180.08	10.43	95.26	187.48	10.63	99.05	219.83	10.47	115.14
0.268	2705.68	10.84	1358.26	2814.73	11.05	1412.89			

Table 4. Comparison of the QNA and SL models for the uncoded transmitter-based code spread-spectrum slotted-ALOHA multiple-access. The 3-node network in Figure 6 is used. η = 13 dB, E_b/N_0 = 30 dB, $JSR_1 = JSR_2 = 0$, $JSR_3 = -7.3$ dB, $u = v$.

Throughput	Packet Delay								
	SL			QNA			Simulation		
	Node 1	Node 2	Network	Node 1	Node 2	Network	Node 1	Node 2	Network
0.02	44.81	5.47	25.14	49.92	5.48	27.70	54.24	5.54	29.94
0.04	82.39	6.04	44.21	101.24	6.07	53.65	118.98	6.09	62.50
0.06	583.16	6.77	294.96	783.94	6.81	395.38			
0.062	1518.79	6.85	762.82	2059.38	6.90	1033.14			

Table 5. Comparison of the QNA and SL models for the uncoded transmitter-based code spread-spectrum slotted-ALOHA multiple-access. The 3-node network in Figure 6 is used. η = 13 dB, E_b/N_0 = 30 dB, $JSR_1 = JSR_2 = 0$, $JSR_3 = -6$ dB, u = v.

Throughput	Packet Delay								
	SL			QNA			Simulation		
	Node 1	Node 2	Network	Node 1	Node 2	Network	Node 1	Node 2	Network
0.002	148.12	5.14	76.63	157.03	5.15	81.09	167.63	5.19	86.89
0.004	173.85	5.30	89.57	194.78	5.30	100.04	213.41	5.30	109.45
0.008	266.95	5.63	136.29	331.27	5.64	168.46	394.97	5.70	200.83
0.012	577.64	6.02	291.83	786.67	6.03	396.25			
0.014	1388.47	6.23	697.35	1975.05	6.24	990.65			
0.0148	3173.05	6.32	1589.68	4590.53	6.34	2298.43			

Table 6. Comparison of the QNA and SL models for the uncoded transmitter-based code spread-spectrum slotted-ALOHA multiple-access. The 3-node network in Figure 6 is used. $\eta = 13$ dB, $E_b/N_0 = 30$ dB, $JSR_1 = JSR_2 = 0$, $JSR_3 = -5.5$ dB, u = v.

Finally, we have to mention that the number of time slots required to give good estimate of the mean packet delay by simulation in the jamming environments is significantly longer than that in the non-jamming environments. This is explained in the next section.

4.2 Simulation Models

The simulation program comprises four components, namely:

(1) The external packet generator.

(2) The node transmission simulator.

(3) The spread-spectrum multiple-access channel simulator.

(4) The transit packet handler.

There is a buffer assigned to each node. The newly arriving packet (external or transit) to a node is stacked into its buffer and is served on the first-come-first-serve basis. Each packet is represented by a data block containing the following information:

(1) control flag,

(2) source node identification number,

(3) destination node identification number,

(4) packet serial number,

(5) packet arrival time to the network,

(6) end-to-end average window number,

(7) current node identification number,

(8) packet arrival time to the current node,

(9) current node average window number,

(10) hop count.

The control flag determines the type of the packet. The current node is the node where the packet is waiting for processing. The end-to-end average window number and the current node average window number are used to classify the packets. The simulation output is

presented to the user by averaging over the packets in the same class. This feature is useful for simulating the networks under the dynamic jamming attack, where the network behavior changes from one observation window to another observation window. This idea is further explored in Section 5. For the static jamming attack, we are only interested in the long-term average; thus these two entities are not needed. The hop count is the number of nodes being traversed by the packets, including the current node.

The external packet generator consists of two types of random number generators, namely:

(1) Uniform random number generator: If the external arrival rate λ_i is larger than 0.1, the uniform random number generator is called every slot to determine whether a new packet should be created.

(2) Geometrical random number generator: If the external arrival rate λ_i is less than 0.1, the geometrical random number generator is used. At the moment that a new packet is created, this generator is called to determine the next time slot, in which the next new packet should be created.

The combining use of the above two types of random number generators can maximize the simulation speed.

The node transmission simulator calls the uniform random number generator for every busy node in each slot to determine whether the nodes should transmit in the slot. In the subsequent discussion, the result is referred to as the transmission profile which is used in the spread-spectrum multiple-access channel simulator to determine the multiple-access interference level.

The first stage of the channel simulator is to determine whether the target nodes want to transmit. This is done by examining the transmission profile. Only the target nodes which are in the receiving mode are considered in the second stage. The second

stage of the channel simulator is different for the transmitter-based code and the receiver-based code cases.

Transmitter-Code Case:

The packets heard at each node are identified from the transmission profile and a uniform random number is generated to determine which packet should be considered. This process takes into account the factor (1/k) in equation (4.4). When a packet is considered by a node it is not targeted for, the packet is discarded and there is no need to go the third stage of the channel simulator.

Receiver-Code Case:

The packets heard by each node are identified from the transmission profile. The packets targeted at each node are found from the packets heard by it and a uniform random number is generated to determine which packet should be considered. This process takes into account the factor (1/n) in equation (4.10).

At the third stage of the channel simulator, the jamming status at each node is determined by a uniform random number. If node j is jammed, $P_{A_j}(k,J)$ is computed (equation (4.16)); otherwise, $P_{A_j}(k,NJ)$ is computed, where k is the number of packets heard at node j. A uniform random number is generated and is compared to $P_{A_j}(k,J)$ or $P_{A_j}(k,NJ)$ to determine whether the packet should be received correctly. This completes the function of the spread-spectrum multiple-access channel simulator.

The transit packet handler performs the book-keeping work. It removes the successfully received packets from the buffers of the transmitting nodes. The service time of these packets by their respective nodes is recorded properly. If the successfully received packets have arrived at its final destination, their end-to-end packet delay is recorded properly. If the successfully received packets have to be forwarded to the next nodes, they are stored in the respective buffers.

An important problem in the network simulation is to determine how many time slots are needed to produce the good estimates of the desired parameters. Let us first

introduce the concept of the busy periods and idle periods. Each busy period of a queue consists of all consecutive time slots in which the queue is not empty. On the other hand, each idle period consists of all consecutive time slots in which the queue is empty. Since the queue always alternates between the busy and idle periods, we can define a cycle of the queue to be a busy period followed by an idle period. For a single G/G/1 queue, the statistics in two cycles are mutually independent. Therefore, the number of cycles being run through can be a good indicator about the quality of the estimates. For the network of queues, the problem is more complicated since the statistics in the adjacent cycles can be dependent due to the interactions between the neighbor nodes. Nevertheless, we can still use it as the rough indication about the estimate quality.

5. DYNAMIC JAMMING ATTACK

The dynamic jamming attack is more harmful to the networks with the adaptive routing algorithms than the static jamming attack, since the latter can be detected by the communicators and the appropriate actions can be taken to alleviate its effectness. In the subsequent discussion, we first examine the basic characteristics of the adaptive routing algorithms. The simulation models for the dynamic jamming attack is examined next. Finally, the simulation program is illustrated in detail. The analytical models for the dynamic jammed network are not covered in this report.

The adaptive routing algorithms comprise three components, namely:

(1) channel quality monitoring,

(2) routing information exchange and routing table setup,

(3) transmission of data packets.

The detection of the jammer existence is through the channel quality monitoring. A typical method, which is used in the current PRnet technology [6], is to observe the percentage of bad packets in every T_{cm} slots of interval, referred to as the channel monitoring interval (CMI). If the percentage of bad packets at a node exceeds the threshold, this node is declared being jammed. The selection of threshold is important. If the threshold is small, many false alarms may occur and the adaptive routing algorithms can be paralyzed due to many transient loops created by the routing algorithms. If the threshold is large, the communicators may not be able to detect the jammer existence and the adaptive routing algorithm is useless again. When a node is jammed, the in-coming data link is surely bad. However, if the in-coming acknowledgement link is still good (this situation is possible if the acknowledgement packets are protected heavily by a low rate error-correction code), the out-going data link can still be useable. Therefore, depending on the acknowledgement scheme, a jammed node may declare both the in-coming and out-going links bad or it many only declare the in-coming links bad.

The information regarding the link quality and the current routing table at each node are exchanged between the neighbors in every T_{rie} slots of interval, referred to as the routing information exchange interval (RIEI).

Three observations can be made about the CMI and RIEI:

(1) The CMI and RIEI at the different nodes are not necessary to be synchronous. In this simulation study, we assume that they are synchronous throughout the network.

(2) The CMI and RIEI can have different length. However, in order to disseminate the link quality information as soon as they are available, T_{rie} should be equal to or less than T_{cm}. Note that, for each change of network connectivity, it may take many RIEIs to produce a loop-free network routing table; thus it may be advantageous to have $T_{rie} < T_{cm}$. In this simulation study, however, we assume $T_{rie} = T_{cm}$.

(3) Since most of the distributed adaptive routing algorithms create loops in the network routing table (i.e., the end-to-end routes do not exist for some source-destination pairs) during the transient period, it is important to minimize the false alarm probability. Thus the selection of T_{cm} must be adequate.

An important feature of the dynamic jamming attack is its time-varying nature. The networks must monitor the existence of the jammers and react appropriately. Three phenomena in the network operation under the dynamic jamming play important roles to the network anti-jamming performance:

(1) The packets pile up in the jammed area before the jammer is detected.

(2) The packets pile up in the area surrounding the jammed area because of the transient routing loops created by the adaptive routing algorithms.

(3) Inefficient use of the network capacity in the previously jammed area.

Since these three phenomena occur when the jammer status changes, the packet delay statistics changes from time to time in the dynamic jamming environment. More precisely, the packet delay statistics are a function of the packet arrival time relative to the jamming time frame. It is possible that the packets arriving at a node (or entering the network) in a particular time interval suffer the excessive delay. Although the long-term average packet delay can still be a good overall performance measure, it may be misleading in many cases where the worst case performance is concerned. In order to illustrate the aforementioned concept, let us consider the simplest form of dynamic jamming, namely, the periodic fixed block jammer (PFBJ), which was described in Section 3.2. The network behavior in this type of jamming environment is a random process because of the randomness of the jamming strategies and the network response (i.e., the network responds differently in each period of PFBJ). However, the average behavior exists and it is a function of time. In order to obtain the average behavior by simulation, we partition the time into small intervals, referred to as the averaging window. The averaging process is explained in Figure 7. The statistics of the observables are collected window-by-window. The size of the window determines the resolution. If the window size is small, the number of simulation periods required to get the good estimate can be excessive long. Using this technique, we are able to assess the average dynamic network response to the periodic fixed block jammer in the time domain.

The simulation program for the network under the PFBJ is the extension of that under the static jamming attack described in Section 4.2 by adding the following three features:

(1) Channel monitoring: The percentage of bad packets is observed at every node in every CMI. If the threshold is exceeded at a node, this node is declared being jammed. All in-coming links of this node are also declared bad and are not usable.

(2) Routing information exchange: Each node maintains a local routing table which contains the lengths of the routes, i.e., the numbers of links to be traversed, to other nodes. The results of the link quality measurement at each node are incorporated into the local routing table as soon as the information are available. In each RIEI, every node broadcasts its local routing table as well as the quality of its incoming links to all its neighbors.

(3) Averaging window approach: The network and the periodic fixed block jammer are running continuously and the statistics of the packet delay and the queue size are collected in the way illustrated in Figure 7. At the end of simulation, the data are averaged window-by-window and the results are presented as a function of time.

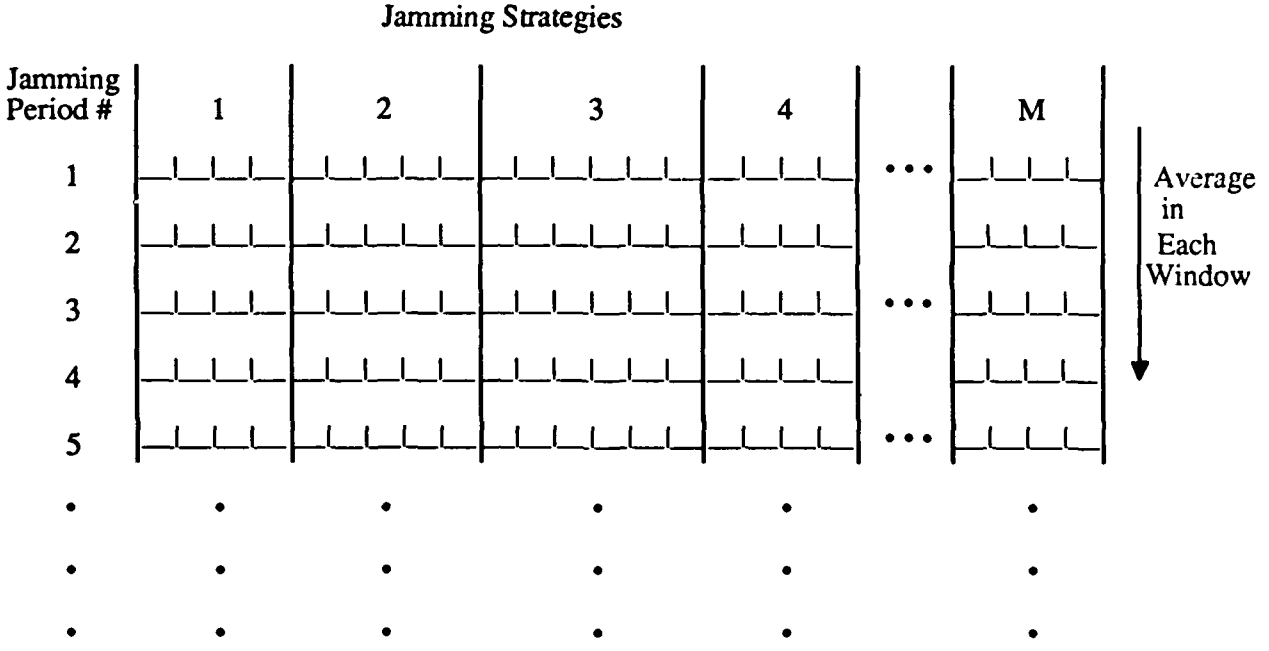


Figure 7. The averaging window approach to the simulation of the network under the periodic fixed block jamming attack.

6. References

[1] J. A. Silvester and I. Lee, "Throughput/Delay Performance of Multi-Hop slotted ALOHA Networks," Technical Report CSI-83-09-01, University of Southern California, September 1983.

[2] A. Segall, "The Modelling of Networks with Radio Links as Applied to C^3 Problems," in Proceedings of the Second MIT/ONR Workshop on Distributed Communication and Decision Problems, Naval Postgraduate School, Monterey, California, 1979.

[3] A. Segall and A. Sidi, "The Modelling of Packet Radio Networks," Technical Report E. E. PUB 356, Technion Israel Institute of Technology, June 1979.

[4] S. L. Su and V. O. K. Li, "An Iterative model to Analyze Multi-Hop Packet Radio Networks," in Proc. Allerton Conf., October 1985. pp. 545 – 554.

[5] W. Whitt, "The Queueing Network Analyzer," The Bell System Technical Journal, Vol. 62, No. 9, November 1983, pp. 2779 – 2815.

[6] J. Jubin, "Current Packet Radio Network Protocols," Proceedings of INFOCOM '85, pp. 86 – 92, April 1985.

[7] J. Westcott, J. Burruss, and V. Begg, "Automated Network Management," Proceedings of INFOCOM '85, pp. 43 – 51, April 1985.

[8] N. Shacham and J. D. Tornow, "Future Directions in Packet Radio Technology," Proceedings of INFOCOM '85, pp. 93 – 98, April 1985.

END

FILMED

MARCH, 1988

DTIC